Quarterly Progress Report

N01-NS-1-2333

Restoration of Hand and Arm Function by Functional Neuromuscular Stimulation

Period covered: July 1, 2001 to September 30, 2001

Principal Investigator: Robert F. Kirsch, Ph.D.

Co-Investigators:

Patrick E. Crago, Ph.D.
P. Hunter Peckham, Ph.D.
Warren M. Grill, Ph.D.
J. Thomas Mortimer, Ph.D.
Kevin L. Kilgore, Ph.D.
Michael W. Keith, M.D.
David L. Wilson, Ph.D.
Joseph M. Mansour, Ph.D.
Jeffrey L. Duerk, Ph.D.
Wyatt S. Newman, Ph.D.
Harry Hoyen, M.D.
John Chae, M.D.
Jonathon S. Lewin, M.D.
Richard Lauer, Ph.D.

Case Western Reserve University
Wickenden 407
10900 Euclid Avenue
Cleveland, OH 44106-7207
216-368-3158 (voice)
216-368-4969 (FAX)
rfk3@po.cwru.edu

Contract abstract

The overall goal of this contract is to provide virtually all individuals with a cervical level spinal cord injury, regardless of injury level and extent, with the opportunity to gain additional useful function through the use of FNS and complementary surgical techniques. Specifically, we will expand our applications to include individuals with high tetraplegia (C1-C4), low tetraplegia (C7), and incomplete injuries. We will also extend and enhance the performance provided to the existing C5-C6 group by using improved electrode technology for some muscles and by combining several upper extremity functions into a single neuroprosthesis. The new technologies that we will develop and implement in this proposal are: the use of nerve cuffs for complete activation in high tetraplegia, the use of current steering in nerve cuffs, imaging-based assessment of maximum muscle forces, denervation, and volume activated by electrodes, multiple DOF control, the use of dual implants, new neurotization surgeries for the reversal of denervation, new muscle transfer surgeries for high tetraplegia, and an improved forward dynamic model of the shoulder and elbow. During this contract period, all proposed neuroprostheses will come to fruition as clinically deployed and fully evaluated demonstrations.

Summary of activities during this reporting period

The following activities are described in this report:

- *Measurement of human upper extremity nerve diameters and branch-free lengths*.
- *Sensor development for arm and hand control in high tetraplegia.*
- *A robotic facility for evaluation of control sources to restore arm function via FNS to individuals with high level tetraplegia.*
- *Rapid prototyping and real-time control for functional electrical stimulation (FES).*
- *Re-innervation of denervated muscle for electrical activation.*
- *Protocols for characterizing myoelectric signals expected during hand grasp tasks.*
- *Feed-forward ANN-based real-time controller for a hand/wrist neuroprosthesis.*

Measurement of human upper extremity nerve diameters and branch-free lengths.

Contract sections:
E.1.a.i Achieving Complete and Selective Activation Via Nerve Cuff Electrodes
E.2.a.i Selective Activation of Elbow and Shoulder Muscles by Nerve Cuff Electrodes

Abstract

The ability to selectively activate peripheral nerve trunk fascicles using nerve cuff electrodes is well established. In the effort to combine several upper extremity functions into a single neuroprosthesis, it is desirable to use this technology for activation of specific muscles. Studies of the anatomy and fascicular structure of the upper extremity nerves are necessary to identify implant sites that maximize success. The anatomical studies include measurements of the diameters and branch free lengths of the nerves on four complete brachial plexuses. The use of lipophilic dyes to map the fascicular studies has been researched and though current reported

maximal tracing distances are inadequate for our purposes, the use of fresh tissue or electric fields to drive the diffusion of the dyes is being pursued.

Introduction

Selective activation of individual fascicles of a peripheral nerve has important potential uses in an upper extremity neuroprosthesis. A nerve cuff electrode could be implanted on the suprascapular nerve (selective activation of the infraspinatus and supraspinatus), the musculocutaneous nerve (selective activation of the biceps, brachialis, and coracobrachialis), and the radial nerve (selective activation of the medial and lateral heads of triceps and the wrist extensors ECRB, ECRL, and ECU). Nerve cuff electrodes could also be used to activate muscles with complex shapes and innervation patterns where muscle-based electrodes would be difficult to implement. In the upper extremity potential, sites for this application of nerve cuff electrodes include the long thoracic nerve (serratus anterior), axillary nerve (deltoid, teres minor), thoracodorsal nerve (latissimus dorsi), and the subscapular nerves (subscapularis, teres major).

To maximize the probability that the nerve cuff electrodes will enable selective activation of the desired muscles, it is necessary to study the anatomy and fascicular structure of the upper extremity motor nerves. Cadaver dissections of the upper extremity nerves are being used in this study. The anatomical study includes measurements of the diameters and branch-free lengths of the nerves to identify implant sites for multi-contact cuff electrodes.

Further, it is necessary to determine the fascicular structure of the nerves so that accurate computer models can be created to simulate the desired selective stimulation. A pilot study of fascicle mapping by cross-sectioning an excised radial nerve showed that it is extremely difficult to follow the rapidly changing fascicular topography. Once fascicles merge it becomes difficult to identify one branch from another or make predictions on whether distinct branches can be found inside of a fascicle that may contain several branches. The use of lipophilic carbocyanine dyes to trace distinct muscle branches is being pursued to overcome the limitations of cross-sectioning.

Dissection Methods

The cadaver dissections were performed using blunt dissection techniques. The brachial plexus, its branches, and the branches of the axillary, musculocutaneous, and radial nerve were identified and tagged. Any irregularities were noted and photographed. Length measurements were done using calipers, and the diameters of the branches and nerve trunks were measured using a suture to measure the circumference of the nerve and assuming a circular cross section. After these measurements were completed the nerves were excised en masse with muscle branches tagged with sutures for identification. The nerves were then stored for future studies of their fascicular structure.

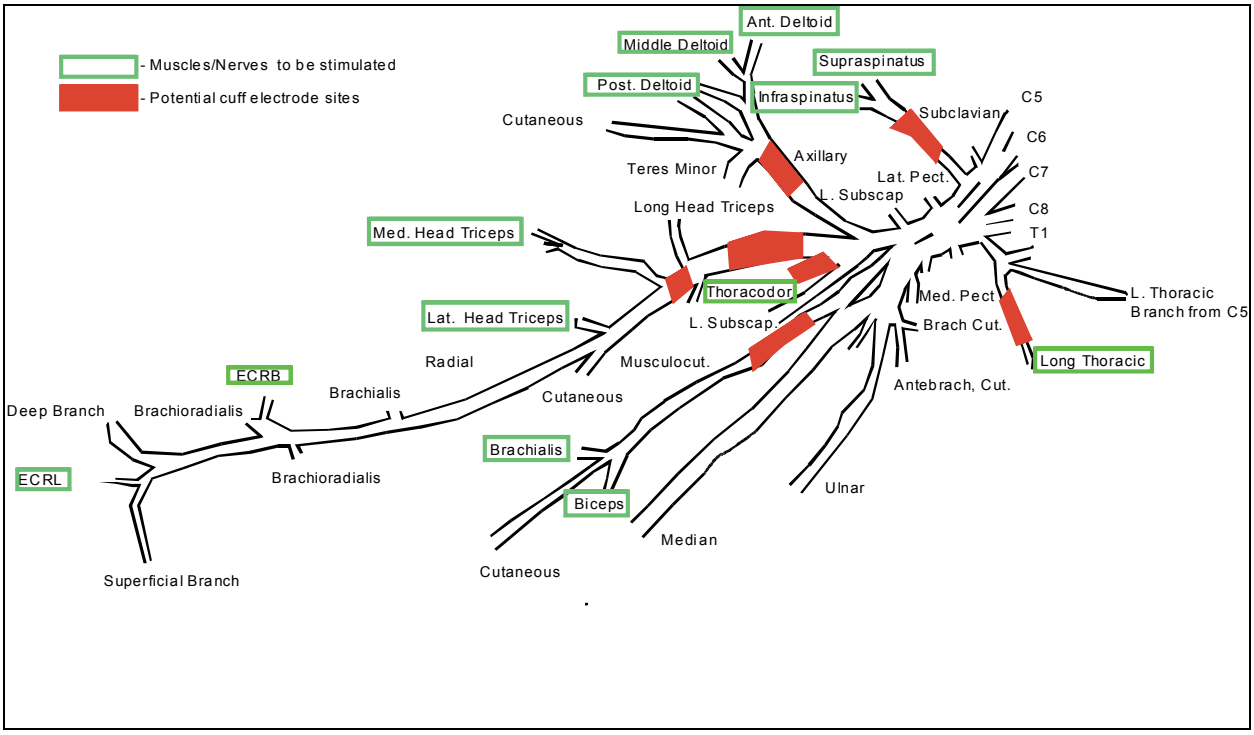

Figure 1. Right upper extremity nerves (9802R)

Dissection Results

Two complete sets of brachial plexuses have been dissected with branch-free lengths and circumference measurements taken. Figure 1 shows one of the complete plexus dissections with the muscles and nerves identified as stimulation targets and likely cuff electrode sites highlighted. Table 1 shows the maximum and minimum circumferences and branch-free lengths for these potential cuff electrode sites. For current multi-contact electrodes to be used a branch free length of at least 2 cm and a diameter of at least 1.6 mm are necessary.

	Branch Free Length (cm)			Diameter (mm)		
Nerve	Min	Mean	Max	Min	Mean	Max
Suprascapular	5.3	5.5	6.1	1.6	1.7	2
Axillary	5.5	5.8	6.4	2.5	2.7	3.2
Musculocutaneous	4.9	5.1	5.5	2	2.4	2.6
Radial (pre-lateral triceps branches)	5.9	6.3	6.7	3.5	3.8	4
Radial (post-lateral triceps branches)	0.9	1.6	2.2	3.3	3.5	3.9

Table 1. Branch-free lengths and circumferences

All of the sites where a multi-contact cuff electrode would be necessary meet the minimum requirements for these electrodes. It would be ideal to place the cuff for the radial nerve below the triceps long head branch but the measurements taken have shown that the

branch free length between the triceps long head branches and the medial triceps branches is not always long enough for current multi-contact cuff electrodes. Measurements where also taken for the long thoracic and thoracodorsal nerves. These nerves provide ample branch free lengths but do not meet the minimum diameter requirements for a multi-contact cuff electrode. Both of these nerves innervate only one muscle though, so single contact cuff electrodes that can be made for smaller diameter nerves could be used.

Fascicle Mapping using Lipophilic Dyes

Retrograde tracing using lipophilic dyes has been used in postmortem aldehyde-fixed nerve fibers (Honig and Hume 1986, 1989; Godement et al. 1987). The main limitations of these dyes has been the limited maximal tracing distance that can be achieved due to the slow diffusion rate in fixed tissue. Lukas et al. found maximal tracing distance for DiI (1,1'-dioctadecyl-3,3,3'3'-tetramethlylindocarbocyanine perchlorate), DiO(3,3'-dioctadecyloxa-carbocyanine perchlorate), and DiA (4-(4-dihexadecylamino)styryl)-*n*-methylpyridinium iodide) for fixed human tissue. Their results (Figure 2) found that incubation at 37 °C for 12-15 weeks provided optimal tracing distances (28.9 ±2.2 mm for DiO).

These maximal distances fall well below the tracing distances that are necessary for creating fascicle maps of peripheral nerves. In the radial nerve tracing distances of at least 40 cm would be desirable to identify muscle branches that are targeted for selective activation at the most likely cuff electrode sites. Two different approaches are being considered to increase the tracing distances of the lipophilic dyes: application of electric fields to create a electromagnetic force that assists diffusion, and use of fresh, rather than fixed tissue.

The first approach hinges upon the fact that a number of these dyes including DiI, DiO and DiA are cationic. It could be possible to place the nerves that are being traced in DC electric fields to increase the force behind the diffusion process. This could speed up the diffusion of the dyes in the nerve tissue and increase the maximal tracing distances that can be achieved. A trial of this theory was done with the dyes placed in a 0.8% agarose gel block. The block was subjected to a 125 V/m electric field for 6 hours. No diffusion of the dyes was noted during this time. However a control block that was left for 21 days also showed no diffusion. This leads us to believe that either the pore size of the gel was too small to allow for diffusion or that the agarose gel environment is not suitable for diffusion of these dyes.

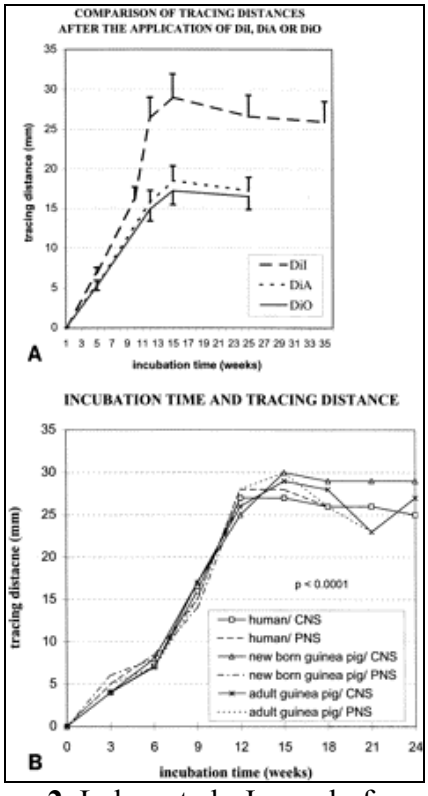

Figure 2. Lukas et al., Journal of Histochemistry &Cytochemistry, 1998.

The second approach is to use unfixed fresh tissue. Sparks et al. have reported diffusion rates of 1 mm/hour for DiI when fresh nerve tissue was used (2000). They were able to reach maximal tracing distances of 40mm in 40 hours before the tissue was then fixed for cross-

sectioning. This diffusion rate is several magnitudes of order greater then any previously reported in fixed tissue. It may be possible to diffuse for longer time spans prior to fixation to increase the maximal tracing distances.

Next Quarter

The dissection portion of this work will be completed with two more complete brachial plexuses dissected. The lipophilic tracing of the nerve branches will be further explored with continued experiments to test the viability of using electric fields to drive the diffusion of the dyes through the nerve tissue. If these experiments yield negative results then the possibility of using fresh tissue for the nerve tracing will be investigated.

References

Godement P, Vanselow J, Thanos S, Bonhoeffer F. A study in developing visual systems with a new method of staining neurons and their processes in fixed tissue. Development 1987; 101:697-713.

Honig MG, Hume RI. Di-I and Di-O: versatile fluorescent dyes for neuronal labeling and pathway tracing. Trends Neurosci 1989;12:333-35.

Honig MG, Hume RI. Fluorescent carbocyanine dyes allow living neurons of identified origin to be studied in long-term cultures. J Cell Biol 1986;103:171-87.

Honig MG. DiI labeling. Neurosci Protocols 1993;50:1-20.

Lukas J, Aigner M, Denk M, Heinzl H, Burian M, Mayr R. Carbocyanine Postmortem Neuronal Tracing: Influence of Different Parameters on Tracing Distance and Combination with Innumocytochemistry. J Histochem & Chem 1998:46(8):901-10.

Sparks DL, Lue L, Martin TA, Rogers J. Neural tract tracing using Di-I: a review and a new method to make fast Di-I faster in human brain. J Neurosci Meth 2000;103:3-10.

Sensor Development for Arm and Hand Control in High Tetraplegia

Contract Section: E.1.a.iv Command Sources for High Tetraplegia

Abstract

The purpose of this study is to develop a user interface to an advanced neuroprosthetic system that will be used to restore upper extremity function to individuals with high tetraplegia resulting from a spinal cord injury at or above the fourth cervical level (C3-C4). These individuals retain voluntary control over the muscles of the head and neck, but have complete paralysis of both upper and lower extremities. The goal of this project is to determine the most appropriate method of enabling these individuals to control the movement of their arm and hand. The outcome of this project, a neuroprosthetic control algorithm for high level tetraplegia, will be combined with the outcome of the arm movement and coordination projects to produce a complete neuroprosthetic system.

Methods and Results

Work during the last quarter has focused on the development of the high tetraplegia interface using the Polhemus Fastrak sensor to record head orientation, and four recording

electrodes connected to two differential amplifiers to record eye movement (electro-oculogram or EOG) and cranio-facial muscle contractions (EMG). The locations of the recording electrodes were vertical across the right eye, and horizontal across both eyes. This placement allowed for the recording of eye blink information, and right/left and up/down gaze. The recording pair vertically placed across the right eye also recorded the EMG activity of the frontalis and the temporalis muscles.

The testing of the recording sensors was accomplished using an interface developed in the LabVIEW programming language. The LabVIEW interface was designed to simulate what would be required from these sensors in neuroprosthetic control. Eye blink was used as a switch to initiate and stop the recording of data from the Fastrak sensor. This is similar to the algorithm set forth in the proposal, which called for three successive eye blinks to initiate a change in the neuroprosthesis. However, three eye blinks were difficult to differentiate from an EMG signal generated by the cranio-facial muscles, even with low pass filtering. This situation was rectified by using two rapid eye blinks, with rectification of the signal, to provide for the switch to initiate changes in the system. This change to two eye blinks from three should not affect the ability to operate the neuroprosthesis.

The EMG signal, also recorded from the right eye pair, was used to activate a series of switches that represented command changes to open and close the hand. The algorithm set forth in the proposal called for frontalis activation to control hand opening and temporalis activity to control hand closing, with a brief burst of activity to initiate a lock of the hand. However, given the location of the recording pair it was difficult, without the introduction of a neural network or another type of classifier, to differentiate between the activities of these two muscles. Therefore, the LabVIEW program was constructed to work on a tri-level basis. Activation of a muscle above or below the first level toggled the switch to initiate hand opening and closing. A higher level burst of activity initiated the hand lock. Again, this modification of the algorithm should not present any problems for neuroprosthetic operation.

The Fastrak sensor, once the command was given through the eye blink, was responsible for cursor movement upon the screen. Movement of the head up and down moved the cursor in the y-direction, and head rotation was responsible for x-direction movement. Head tilt (side to side movements) controlled a dial placed above the cursor movement grid. This is identical to the algorithm that was set forth in the original proposal. There was a slight degree of cross talk between the three movements, however this was expected since true movement of the head in only one direction is not possible. This can be rectified by high pass filtering of the signal, which may be desirable to do in hardware given the computational requirements of the interface.

The ability to record right and left eye movements, while possible, was not programmed into the testing software. The differential amplifiers used with the electrode pairs were originally designed for EMG recording. Therefore, the capability of recording eye gaze information appeared to be limited to transient information, implying that only eye blink and eye glances could be detected. The continuous tracking of eye position, as required in the algorithm proposed in the contract did not appear to be possible. Therefore, a possible modification of the algorithm may be to have an individual quickly glance to the right or left, depending upon the direction they wish to move, which will act as a switch to move in that direction. Head movement in that direction will then move the arm. Cessation of movement would require a quick glance in the other direction. Considerations will need to be made, however, on whether the side to side eye gaze will be useful for the control algorithm, especially since the recording of this movement can

only be accomplished by an electrode pair placed horizontally across the eyes. This differs from the original proposal of the interface, which only stated an electrode pair above the eyes.

Next Quarter

During the next quarter, work will begin on interfacing these sensors with a robotic arm to test their ability to control a moving system. In addition, work will be performed on re-evaluating the control algorithm as proposed in the contract for control over the high tetraplegia neuroprosthesis. This will involve investigations into other methods of extracting information from the EMG/EOG recording pair, including wavelet decomposition and neural network classification.

A Robotic Facility for Evaluation of Control Sources to Restore Arm Function via FNS to Individuals with High Level Tetraplegia

Contract section: E.1.a.iv Command sources for high tetraplegia

Abstract

The objective of this portion of the study is to use a robotic manipulator to emulate the kinematics and dynamics of a human arm under FNS control. The intent is to be able to emulate candidate controllers prior to the implementation of a full neuroprosthesis in individuals with high tetraplegia, enabling faster evaluations of a larger number of proposed controllers using able-bodied subjects. Applicability, acceptability and practicality of candidate control algorithms are to be evaluated in a robotic testbed. This facility will enable rapid, quantifiable and repeatable measurements on the performance of alternative control propositions, as well as subjective evaluations for determining the desirability of a control option. The testing will be accomplished by having the users control a robotic manipulator to perform the same tasks that are targeted for neuroprosthetics. At the completion of this project, the most successful control algorithm will be selected and used in the first implementation of an implanted neuroprosthesis for high tetraplegia.

Methods

Development of the robotic facility requires efforts in four areas: mechanical design of appropriate support and safety structures; software design for robot safety; electronic/control-system interfacing; and control software development for emulation of human-arm dynamics.

Mechanical Design

A mounting stand is required to mount the robot—a Kawasaki JS-10—in a position and orientation that can provide reasonable emulation of human-arm kinematics. In addition, a safety shield is required, enabling an operator to exert control over the robot while positioned within reach of the robot, emulating an approximation of natural-arm motion relative to the operator.

In the previous quarter, it was established that the Kawasaki JS-10 robot could exert sufficient torques to be mounted horizontally. During this quarter, a suitable stand and operator safety shield were designed, detailed and are currently being fabricated.

Figure 3 shows a side-view scale drawing of the stand with the robot mounted. The robot base (first axis) is horizontal, mounted approximately shoulder high relative to a standing person (59" from the floor). At this height, the robot arm cannot reach the floor. Thus, if an emergency stop were hit, cutting power to the robot, there would be no damage to the equipment. Also, if a human operator is standing up, he/she can approach closer to the base of the robot, more realistically emulating the relative kinematics of a natural arm.

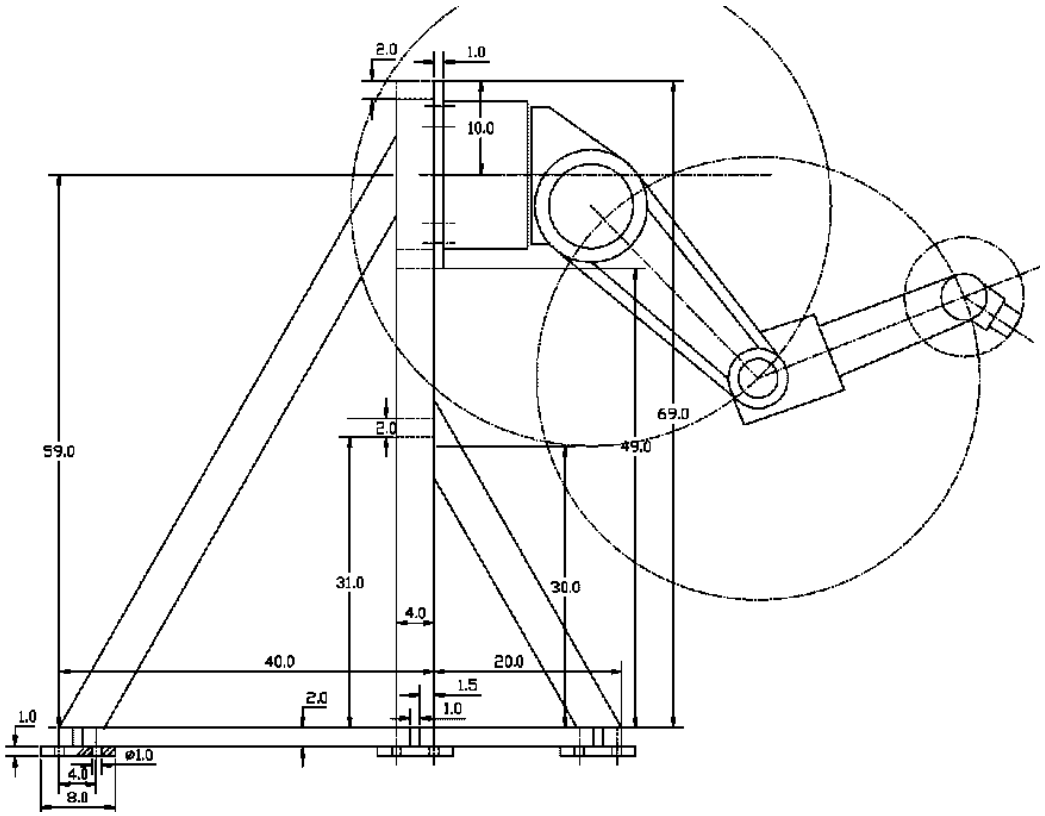

Figure 3: Side view, scale drawing of robot test stand. Construction is of welded tubular steel beams. The Kawasaki JS-10 robot is shown mounted horizontally, providing motion capability similar to a human arm. Under E-stop conditions, the robot can safely fall towards the floor without risk of collision.

Figure 4 shows a front view of the test stand and safety shield. The safety shield consists of Lexan panels supported by welded steel beams. A mock-up of this safety shield is being fabricated for testing. It is predicted that the safety shield is an order of magnitude stronger than necessary to protect against a collision by the robot approaching at maximum speed and exerting maximum torque. A destructive test will be performed to confirm a conservative safety factor.

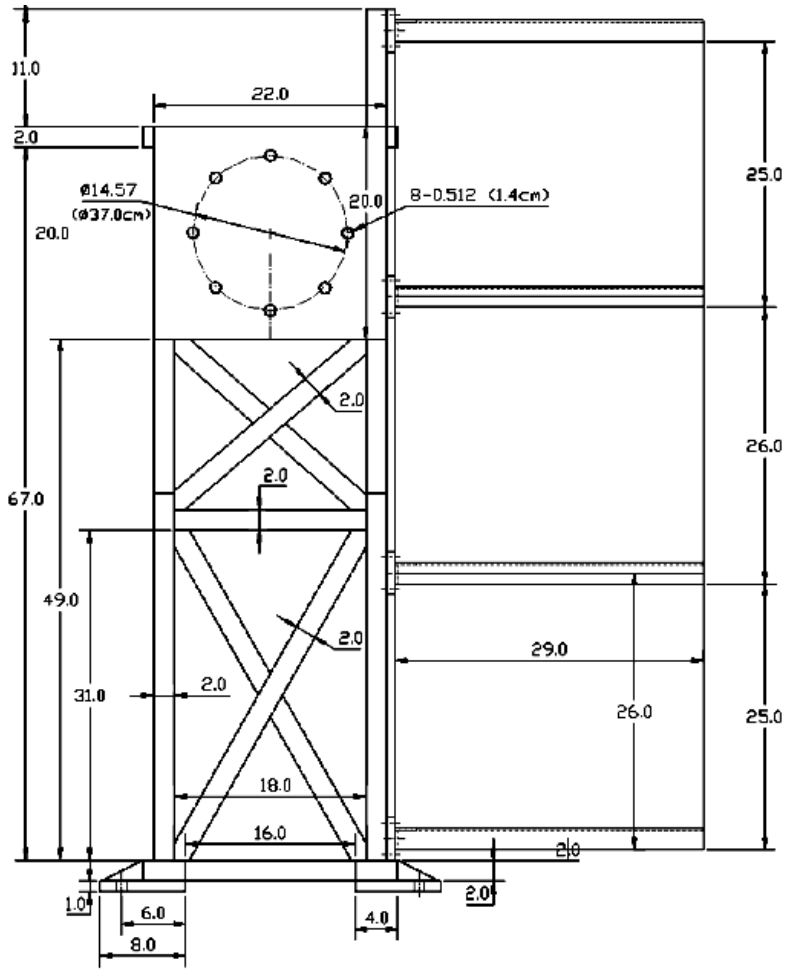

Figure 4: Front view of robot test stand. The safety shield to the right consists of steel support beams and Lexan panels. The operator would stand behind this shield, permitting approximately alignment of the robot's "shoulder" with the operator's shoulder. The robot could approach the operator's face, but the Lexan shield would prevent contact between the robot and the operator.

The operator would stand behind this shield while controlling the robot. The robot will be able to approach close to the operator's face on the opposite side of the shield. It should be possible approximate face washing by having the robot wash the shield in front of the operator. Similarly, feeding could be approximated by having the robot bring food close to the operator's face.

Software design for safety

A second effort this quarter has been the design of safety software. Although the Lexan shield will protect a human operator from the robot, it is nonetheless desirable to prevent the possibility of collisions. Collision-avoidance software is being built into the controller. Figure 5

shows the structure of this design. If user commands are allowed to control the robot directly, then the robot could be accidentally driven into a damaging state. In the controller under development, the state of the robot, a model of surrounding obstacles and self-collision poses, and the user commands are considered in a module that combines these influences. A virtual repulsive force, repelling the robot from mapped obstacles, is added to the user input, resulting in a virtual compliance protecting the robot from high-speed impacts.

During this quarter, this concept was tested in a simplified form. Based on encouraging results, this approach will be developed and utilized.

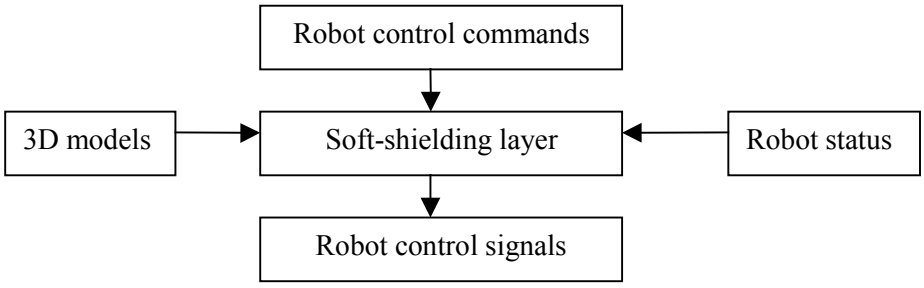

Figure 5: Software architecture for collision avoidance module. Influences from both operator inputs and virtual repulsions from mapped obstacles are summed to produce a net command signal.

Electronic/control-system interfacing

An essential task in constructing the robotic test facility is the ability to control the robot arm from EMG signals. This interface must be performed at a level suitable for emulating human-like arm dynamics and for easily inserting alternative neuroprosthetic control modules for testing. During this quarter, we established and tested a physical interface between EMG signals from the cranio-facial area and the robot controller. This interface, as described in Figure 6 below, senses, amplifies and filters EMG signals using a commercial EMG amplifier (CED-1902), samples these analog signals via analog-to-digital interface modules installed within the robot control computer, and makes these signals available to the robot control software. Using an open-architecture control platform, we have been able to modify the robot controls to respond to these EMG signals. While we may introduce variations in the future, this physical interfacing task is considered successfully concluded.

From this point, attention will focus on placement of electrodes and on interpretation of signals. EMG signals from forehead and upper cheek muscles under each eye were picked up as

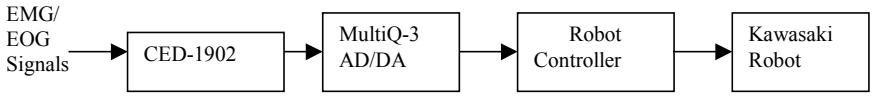

Figure 6: Interfacing of EMG/EOG signals to robot control system. These modules have been successfully integrated.

three independent channels of control inputs. Experiments showed that a human subject can control the robot to move in Cartesian space in velocity control mode. However, it is already apparent that muscle fatigue and channel cross-talk will be issues in designing a control algorithm for a neuroprosthesis. Using the same system, experiments are being conducted to record electrooculographic (EOG) signals generated by eye movements. Both lateral EOG and vertical EOG signals showed encouraging resolution and repeatability. EMG signals due to eye blink are also being tested for possible use as control-mode switching input (see "Sensor development for arm and hand control in high tetraplegia" above).

Control software development for emulation of human-arm dynamics

This task was initiated late this quarter. It will be the focus of efforts over the next quarter. In this task, the robot controller is being redesigned to emulate natural arm dynamics. To do so, we will include a dynamic model of the human arm and stimulate that model in real-time simulation using the EMG/EOG signals per a candidate control algorithm. Through this real-time simulation, the simulated arm dynamics define an equivalent virtual-reality model of the hypothetical arm/control dynamic interaction. The real-time outputs of this virtual arm model are to be used as real-time inputs to a stiff trajectory-tracking control system for the robot. Thus, the robot will produce the simulated dynamics, emulating realistic, human-like arm motion, as driven by the neuroprosthetic control algorithm under test. It will be important to make this emulation as faithful as possible, so that tests with this system will yield valid results in evaluating candidate neurocontrollers.

We anticipate the need to incorporate force/torque sensing in this design. As depicted in Figure 7, physical sensation of contact forces/torques is to be combined with EMG signals from the operator (as processed by a candidate neurocontrol algorithm). Both of these contributions will be used to drive the dynamic model. The robot will be servoed to follow the dynamic model faithfully. Using this approach, it should be possible to emulated realistic dynamics that include contact forces. This will be necessary for operations such as face washing or stabbing food with a fork.

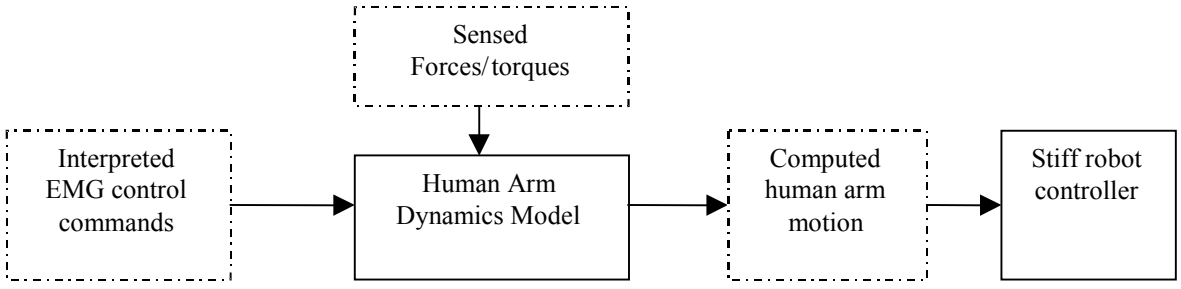

Figure 7: Software plan for achieving human-like dynamics with the test robot. Both command signals from the operator and force/torque signals from a wrist sensor on the robot are used to drive a real-time model of human arm dynamics. The real-time outputs of this model are to be used as trajectory commands to a stiff robot control system.

From previous experience, we anticipate the need to constrain the inertial terms of the dynamic model to be reasonably similar to the actual inertias of the robot. This constraint may make the robot seem less natural for motions in which inertial terms are important (e.g., high accelerations). However, using the robot's inertial model will make contact dynamics stable and responsive. Ultimately, we may choose to use different dynamic models for evaluating contact vs. non-contact tasks.

Results

The robot stand and safety shield are being fabricated. A concept for a software safety module has been tested in primitive form. Encouraging results motivate further development and use. Electronic interfacing has been successfully completed, enabling control of the robot via EMG/EOG signals. The software architecture for the robot controller has been defined, enabling emulation of realistic arm dynamics and testing of candidate neurocontrol schemes.

Next Quarter

In the next quarter, the safety shield will be tested to assure a comfortable safety factor, subject to worst-case robot runaway. The robot stand and safety shield will be fabricated and installed and the robot will be mounted horizontally. A more complete software safety module will be integrated into the robot controller. Additional testing with EMG, EOG and eye-blink measurements will be performed to identify useable signals. An initial, approximate dynamic model of the human arm will be integrated into the robot controller, allowing the robot to imitate human-like dynamics.

Rapid prototyping and real-time control for functional electrical stimulation (FES)

Contract section: E.1.a.vi Implementation and Evaluation of Neuroprosthesis for High Tetraplegia

Abstract

In this project, we are developing a set of software tools to provide rapid prototyping and real-time control for FES systems. The abilities to read physiological data, process and make calculations based on those inputs, then adjust and communicate stimulation parameters at repeatable, well-defined time intervals are essential to control and develop algorithms for neuroprostheses. Typically, such real-time control has been achieved only by computer engineers or programmers with C or assembly language, thus moving the development effort away from the researchers or clinicians who are directly studying the system. By using the much higher level and more abstract means of programming required for software packages from The MathWorks [1-8], we aim to put into the hands of these users, themselves, the abilities to quickly design, implement, test, revise and re-test ideas in real-time. Our tools will provide a general mechanism for studying FES-related problems, and development of application-specific solutions using these tools will be left to individual researchers or clinicians. Therefore, we will be able to centralize the technology development and support for the core software that serves

many projects in this contract. This reporting period represents the initial efforts at developing these tools. Described below are our basic setup and our first deliverable product.

Methods

A typical FES system is shown in Figure 8. In order to provide rapid prototyping and real-time control, we use personal computers (PCs) running MathWorks software in place of the "host" and "target processors," as described below.

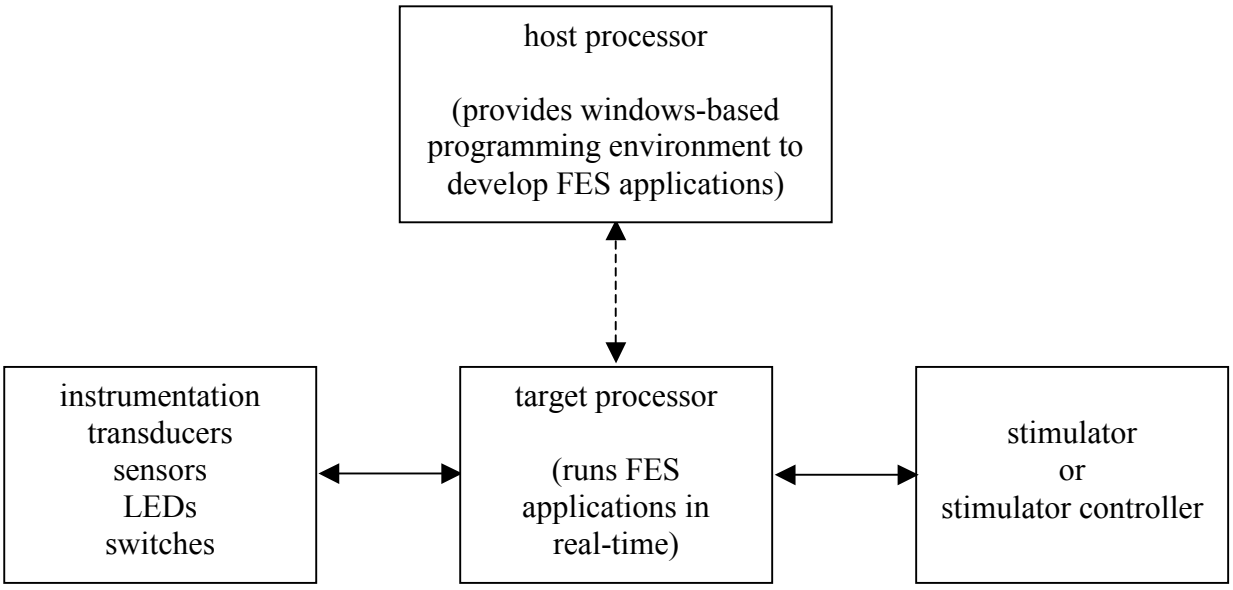

Figure 8. Basic structure of an FES system. The host processor may be present only initially to download the application, or it may always be required to control the target processor.

Software packages from The MathWorks

The following provides a brief overview of the tools that are being used to provide rapid prototyping and real-time control for FES.

In Simulink [1,2], an application running on a personal computer (PC) with a windows-based operating system, block diagrams and state machines can be drawn and then simulated. A multitude of blocks are provided for handling input and output, performing discrete and continuous mathematical functions, creating one's own customized blocks, etc. These blocks can be arranged and connected as desired, then the resulting block diagram can be simulated with different sets of control or input parameters while output signals are monitored. Real-Time Workshop [3,4] can be used to convert a Simulink block diagram into C code that can be executed in real-time on several target processors, such as a second desktop PC.

When a second PC is booted from a floppy disk containing the xPC Target [5,6] kernel and linked to the first PC via a serial or network connection as shown in Figure 9, Real-Time Workshop can convert a Simulink block diagram into C code, then compile, link and download it to the second (target) PC. On the first (host) PC, MATLAB [7,8] is used to control the downloaded application through the serial or network connection, allowing changes in parameters and sample times, starting and stopping of execution and uploading of data, which

can then be further processed or displayed in MATLAB. Changes to the algorithm in Simulink are compiled and downloaded by Real-Time Workshop with just the click of a button so that MATLAB can run the new iteration of the application on the xPC Target machine in real-time.

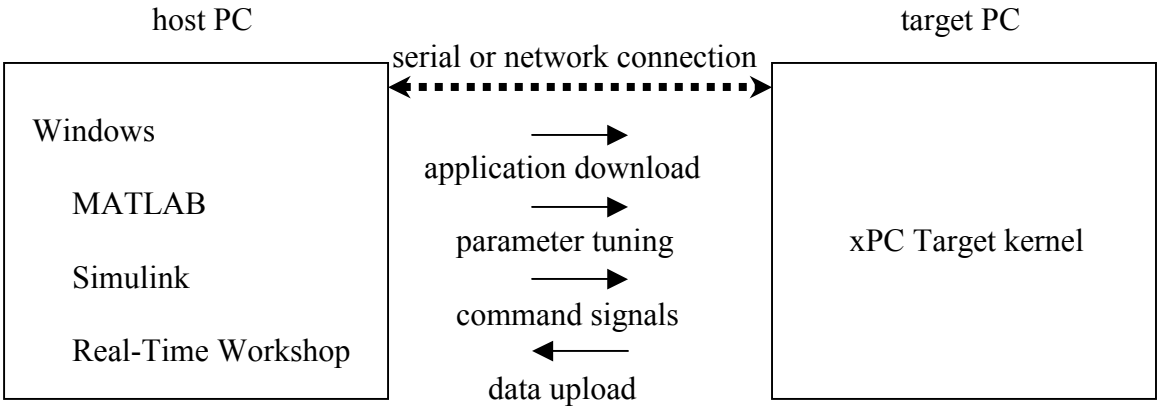

Figure 9. Interaction of The MathWorks packages for rapid prototyping and real-time

Interaction with laboratory equipment using factory-delivered and custom-built software

There are many "I/O device driver" blocks that come with Simulink and are designed to be downloaded and run in real-time on a target PC booted with the xPC Target kernel. Therefore, code is already written and ready to be integrated into block diagrams for interacting with analog-to-digital (A/D), digital-to-analog (D/A), digital input/output, counter, GPIB, serial, etc. interfaces from many different manufacturers. As described below in the "Results" and "Next quarter" sections, A/D and D/A blocks for controlling a specific board have already been tested, and are planned to be further used, as part of a proof-of-concept algorithm for rapid prototyping. Simulink's RS-232 "setup" and "send" blocks for serial communication have been extensively used and modified, as listed below, in order to create a new block for researchers to employ in their algorithms for controlling stimulus pulses.

- Simulink's serial "send" block was designed to transmit character strings, thereby representing a number as the ASCII character code of each of its digits. For example, the number "123" would be interpretted as the string "123" and sent in series as the bytes "49," "50" and "51," which are the ASCII character codes for the digits "1," "2" and "3." In order to allow sending the actual numbers that specified such information as stimulus pulse parameters, the C code underlying Simulink's "send" block was investigated, the commands for sending numerical bytes were isolated and nearly everything else having to do with strings and conversion to ASCII character codes was removed.
- Next, this C code was expanded to allow the user to specify the pulse duration and pulse amplitude as varying inputs and the stimulator channel as a constant parameter. A checksum [9], a one-byte field for error detection, and "SLIP" [10], a packet framing protocol that defines a sequence of characters to frame IP packets on a serial line, were then incorporated in the sending of this message of numerical bytes.
- The user interface for Simulink's RS-232 "setup" block was altered to remove user choices for the baud rate, number of data bits, parity, number of stop bits and protocol,

which were set, respectively, to constant values of "38400," "8," "none," "1" and "none." Then a new block with a new user interface, including "help" documentation, was created to combine the modified "setup" and "send" functions.

- Finally, a script was written to automatically install this new block into Simulink's library so that users would be able connect it into their block diagrams just like any other block.

Results

Using The MathWorks tools, an algorithm was implemented to read data in from an A/D channel of a National Instruments Corporation's multifunction board, multiply these data by a constant, then output the original and scaled data to the D/A channels of the board. The Simulink block diagram representation of this algorithm is shown in Figure 10, and a glimpse of its operation is shown in an oscilloscope printout in Figure 11. This example serves as a proof-of-concept for control of instruments that respond to commands of varying voltage levels; for reading data from sources that are, or can be transduced into, voltages; for rapid prototyping, because any calculation can be substituted for the simple gain operation; and for real-time control, because even at a block diagram sample time, or update rate, of 10 kHz, there is no delay between the system's input and output.

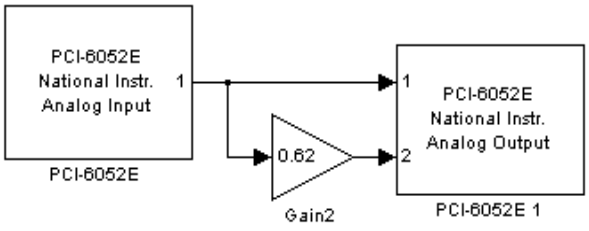

Figure 10. A/D - gain - D/A block diagram in Simulink.

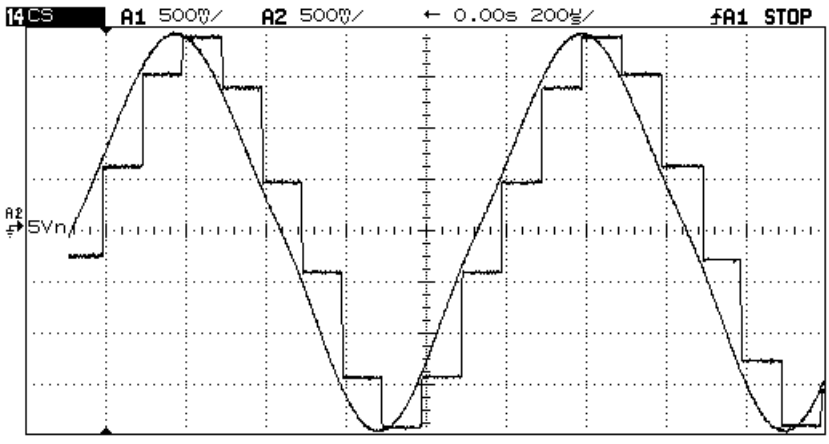

Figure 11. Real-time execution of the block diagram in Figure 8. Plots are the actual signal (2 V sinusoid, 1 kHz) going into the A/D and the output of D/A channel 1.

Figure 12 shows the incorporation of our new block, "one ch stim," or "one channel stimulation," into the Simulink Block Library Browser, and Figure 13 demonstrates use of this block in a very simple algorithm. When this block diagram is run, serial commands are issued from the target PC to evoke stimulus pulses with a given duration and amplitude from a single channel, which is selected by double clicking on the block and entering that parameter. The duration, amplitude and channel can be changed during execution. The pulses continue until the block diagram is interactively stopped or until it reaches its appointed "stop time." One pulse is delivered for each sample period, or update period, of the block diagram, therefore setting the execution rate for the block diagram determines the stimulation frequency.

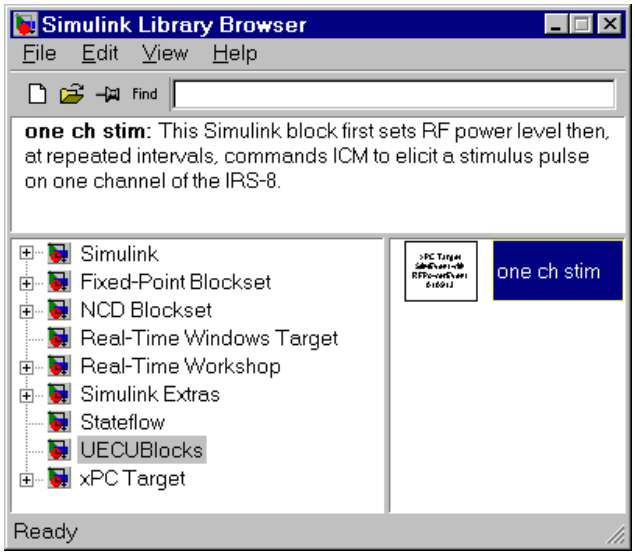

Figure 12. Our "one ch stim" block incorporated into the Simulink Block Library Browser.

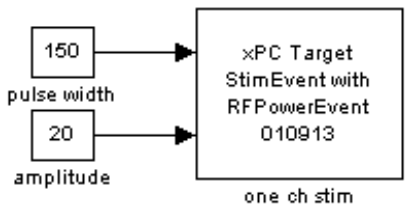

Figure 13. Example block diagram for one channel of stimulation.

Next Quarter

More complicated block diagrams are intended to verify and validate the "one ch stim" block. The pulse duration and amplitude can be calculated instead of simply being specified as constants. For example, they may be determined from processing values read through an A/D channel at a similar, or much faster, rate than the rate of stimulation. Multiple channels may be stimulated together by having multiple copies of the "one ch stim" block in a block diagram, each with a different channel number selected, but the order in which these channels fire needs to be better understood and controlled. Most importantly, the block needs to be distributed and used by researchers in the field so that their feedback can influence further development efforts.

A future block that builds on "one ch stim" will allow stimulation frequency to be updated during algorithm execution. The Simulink block for receiving serial strings will be converted, as the "send" block was in this report period, to read numerical bytes. Other blocks will be created for reading data from stimulators and performing more sophisticated timing control and scheduling of stimulation events so that programmers don't have to specify each individual pulse. Also, efforts will be undertaken to secure regulatory approval for use of this software in human studies.

References
1. Simulink, version 4.0, software release 12, The MathWorks Incorporated, Natick MA, June 2000.
2. Using Simulink (Version 4), The MathWorks Incorporated, Natick MA, 2000.
3. Real-Time Workshop, version 4.0, software release 12, The MathWorks Incorporated, Natick MA, March 2000.
4. Real-Time Workshop User's Guide (Version 4), The MathWorks Incorporated, Natick MA, 2000.
5. xPC Target, version 1.1, software release 12, The MathWorks Incorporated, Natick MA, June 2000.
6. xPC Target Getting Started Guide (Version 1), The MathWorks Incorporated, Natick MA, 2000.
7. MATLAB, version 6.0.0.88, software release 12, The MathWorks Incorporated, Natick MA, October 2000.
8. Using MATLAB (Version 6), The MathWorks Incorporated, Natick MA, 2000.
9. Mallory, T. and Kullberg, A., "Incremental Updating of the Internet Checksum," BBN Communications, Network Working Group Request for Comments: 1141, January 1990. (http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1141.html)
10. Romkey, J., "A Nonstandard for Transmission of IP Datagrams over Serial Lines: SLIP," Network Working Group Request for Comments: 1055, June 1988. (http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1055.html)

Re-innervation of Denervated Muscle for Electrical Activation

Contract section: E.1.c Re-innervation of Denervated Muscle for Electrical Activation

Abstract

The goal of this project is to develop a method for diagnosing denervation in spinal cord injury, and to develop methods for providing re-innervation to denervated muscles. During this quarter, the evaluation of denervation in acute spinal cord injury was initiated.

Methods

Lower motor neuron damage (muscle denervation) is an important limiting factor when applying functional electrical stimulation (FES). We have found that denervation is particularly prevalent in key muscles in high level tetraplegia. It may be possible to use nerve transfer techniques in order to restore innervation to key muscles, but it is necessary to definitively identify denervation within six months post injury in order for these procedures to have a high likelihood of success. Therefore, the first phase of this project is to develop a method for diagnosing denervation within weeks to months post-injury.

The diagnosis tool that we will utilize in this project is the stimulated manual muscle test (SMMT). Electrical stimulation is applied to different muscles of the upper extremity using surface electrodes, and the resulting muscle contraction is graded on a 0 through 5 scale. Muscles graded as either 0 (no response) or 1 (twitch or palpable response only) will be

considered functionally denervated. Muscles in which the individual has voluntary movement will be graded using a standard manual muscle test, and will not undergo further testing.

The specific muscles to be tested are: deltoid, triceps, biceps/brachialis, extensor carpi radialis brevis/longus, flexor digitorum superficialis, extensor digitorum communis, adductor pollicis, ulnar innervated finger intrinsics and abductor pollicis brevis. The deltoid and biceps/brachialis were chosen because they are key muscles that are often denervated in high level tetraplegia. The radial wrist extensors (tested together) were chosen because they are frequently denervated in C5 level SCI, and should provide a considerable number of data points for analysis. The remaining muscles are easily stimulated using surface electrodes, and have denervation rates of approximately 10-20%.

During this quarter, the SMMT test was begun on new and existing spinal cord injured individuals. This testing will take place on both the inpatient and outpatients wards at MetroHealth Medical Center. Each individual will be tested upon admission to the rehabilitation floor (typically within a few weeks of injury). Testing will be repeated at discharge (typically 2 - 3 months post injury), at 6 months post-injury, at one year post-injury, and then at yearly intervals thereafter. Since most spinal cord injured individuals return for regular outpatient visits every six months, relatively good follow-up is expected for the majority of subjects.

Next Quarter

During the next quarter, we will continue to perform the SMMT on individuals admitted to the SCI floor with new injuries. In addition, we will evaluate existing subjects who have undergone the SMMT in the past. This will allow us to establish the consistency of denervation for those who are more than one year post-injury.

Protocols for characterizing myoelectric signals expected during hand grasp tasks

Contract section: E.1.b Control of Grasp Release in Lower Level Tetraplegia

Abstract

The purpose of this research is to develop and evaluate an advanced neuroprosthesis to restore hand function in persons with OCu:5 and OCu:6 (International classification) spinal cord injuries. Through this work we will implement in human subjects a control methodology utilizing myoelectric signals (MES) from muscles synergistic to hand function to govern the activation of paralyzed, electrically stimulatable muscles of the forearm and hand. This work encompasses the following objectives:

1) Demonstrate the feasibility of acquiring myoelectric signals that are appropriate for controlling the neuroprosthesis
2) Develop and test control algorithm options using a simulated neuroprosthesis controller
3) Implement myoelectric control of the hand grasp neuroprosthesis in two subjects with C7 level spinal cord injury and evaluate hand performance

Methods

This quarter was devoted to further development of protocols required for characterizing the myoelectric signals recorded from the wrist flexor (FCR) and extensor (ECR) muscles. The signal characteristics required for controlling the neuroprosthesis using the control algorithms we propose are: 1) noise levels and inadvertent activity must be distinguishable from intended control signals, 2) intended control signals must have repeatable amplitudes, 3) co-contraction levels must be unequal, 4) at least two levels of each MES must be distinguishable, 5) sustained signals must be distinguishable from transient bursts of activity. A protocol for recording noise levels and inadvertent activity while the arm is held in different positions in space was tested. Also a protocol for recording intended control signals from the ECR and FCR under different arm positions and hand loading conditions was tested.

Inadvertent Myoelectric Activity During Different Arm Positions and Orientations

The purpose of this protocol is to provide myoelectric data showing noise levels that should be expected during a hand grasp task. The baseline myoelectric activity in the wrist flexors and extensors may change when the subject extends the elbow or rotates the forearm since the proximal tendons of these muscles cross the elbow. It is necessary to know what baseline levels will be so that the thresholds for detecting intentional control signals can be set above those levels.

Surface electrodes were placed over the FCR and ECR and the myoelectric signals were simultaneously recorded. Two 40-second trials were conducted. A specific posture of the shoulder, elbow, forearm, and wrist was maintained for five seconds followed by another posture for 5 seconds. Thus, a 40-second trial consisted of maintaining eight different arm postures. The eight postures for the two trials are shown in **Table 2**. The first four postures (20 seconds) of both trials are the same.

	Trial 1			Trial 2		
Time segment	Shoulder / elbow	Forearm	Wrist	Shoulder / elbow	Forearm	Wrist
0-5 sec	0° abd, 30° flx / 120° ext (Arm at side)	prone	relaxed	0° abd, 30° flx / 120° ext (Arm at side)	prone	relaxed
5-10	"	"	active extension to neutral	"	"	active extension to neutral
10-15	"	supine	relaxed	"	supine	relaxed
15-20	"	"	active flexion to neutral	"	"	active flexion to neutral
20-25	0° abd, 120° flx / 180° ext (Arm reaching up and forward)	prone	relaxed	90° abd, 0° flx / 180° ext (Arm reaching up and to side)	prone	relaxed
25-30	"	"	active extension to neutral	"	"	active extension to neutral
30-35	"	supine	relaxed	"	supine	relaxed
35-40	"	"	active flexion to neutral	"	"	active flexion to neutral

Table 2. Arm postures maintained for 5 seconds each during two 40-second trials for the purpose of recording noise levels as a function of changes in shoulder, elbow, and forearm position.

Intended Control Signals At Different Arm Positions and Hand Loading Conditions

The purpose of this protocol is to provide myoelectric data showing the intended control signals that should be expected during tasks requiring opening and closing the hand with the hand directly in front of the subject, while reaching up in front of the subject, and while holding a load that produces a flexion torque at the wrist. The intended control signals produced by the wrist flexor and extensors may change in its characteristics when the arm is in these two different postures and when the wrist is loaded in flexion. It is important to know how these intended control signals vary so that control rules can be designed to positively identify the intended signals regardless of arm posture or hand load.

Surface electrodes were placed over the FCR and ECR and the myoelectric signals were simultaneously recorded. Eight trials were conducted. During each trial, the subject was asked to generate a maximum voluntary contraction of the ECR or FCR ten times, holding each contraction for four seconds and resting two seconds between contractions. These contractions were performed while the subject maintained their arm in the first two postures of the first protocol. The forearm was pronated during all contractions. No load was applied to the hand during the first four trials; a two-pound weight was wrapped around the hand, creating a flexion torque at the wrist, during the last four trials. **Table 3** summarizes the eight trials making up this protocol.

Trial	Muscle Contracted	Arm Position	Wrist Flexion Load
1	ECR	at side, prone	No
2	FCR	"	"
3	ECR	reach up forward, prone	"
4	FCR	"	"
5	ECR	at side, prone	Yes
6	FCR	"	"
7	ECR	reach up forward, prone	"
8	FCR	"	"

Table 3. Summary of trials for recording intended control signals during different arm positions and with and without flexion load applied to the wrist.

Results

Inadvertent Activity

The processed myoelectric signals from the ECR and FCR during the two trials of the first protocol are shown in **Figures 14 and 15**, respectively. Approximately every five seconds there is a spike in the signals that corresponds to a change in arm posture. In general, the signals recorded from the ECR are of higher amplitude than the FCR signals. The first twenty seconds of both trials are similar. The segment of data from approximately 5 to 10 seconds corresponds to active wrist extension to bring the wrist to neutral against gravity. This produces the highest-level signals from both muscles during the first 20 seconds of both trials. The noise levels during the first twenty seconds (when the arm is not extended), are at about 0.01 in the FCR and 0.02 in the ECR. However, there is a large spike reaching approximately 0.15 in the ECR recording when the forearm is supinated from its prone position (at approximately 10 seconds). The noise

levels are greater when the arm is reaching up both forward (trial 1) and to the side (trial 2). This can be seen by comparing the last twenty seconds of each trial with the first twenty seconds. In both reaching postures, the greatest noise levels occur in the ECR when the arm is supinated (30 to 35 sec), spiking to beyond 0.15 and leveling off at around 0.05 in both trials.

In conclusion, the inadvertent FCR noise reaches levels of approximately 0.01. The inadvertent ECR noise reaches levels of approximately 0.05 to 0.10 when the arm is reaching upward and the forearm is supinated. Spikes in the ECR reach as high as 0.15 when the forearm is supinated while reaching upward.

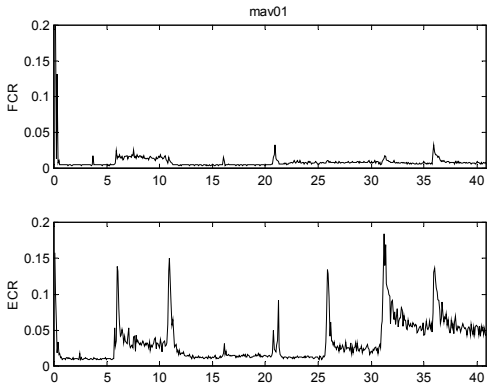

Figure 14. Trial 1 from the first protocol. During the first 20 seconds the arm is at the subject's side with the elbow bent. During the last 20 seconds the arm is reaching up and forward.

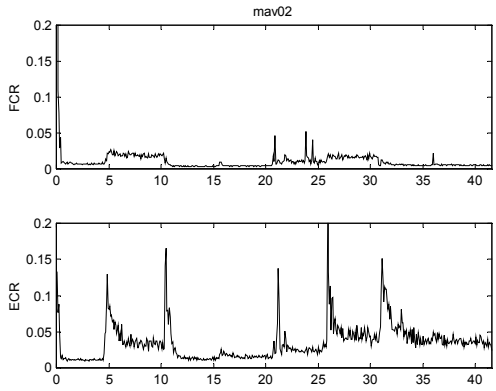

Figure 15. Trial 2 from the first protocol. During the first 20 seconds the arm is at the subject's side with the elbow bent. During the last 20 seconds the arm is reaching up and to the side.

Intended Control Signals

The ranges of contraction levels for each trial from the second protocol are shown in **Figure 16**. The four box plots on the left show the ECR contraction levels, and the box plots on the right show the FCR contractions. With no load on the hand (trials 1 and 3), the ECR contraction levels appear to be fairly consistent regardless of the arm position. When the hand was loaded (trials 5 and 7), the ECR contraction levels increased, and more so with the arm at the side (trial 5) than with it reached upward (trial 7). The FCR contraction levels varied little across the different arm positions and loading conditions, with the highest contraction levels when the arm was at the side and the hand was unloaded (trial 2) and the lowest contraction levels when the arm was reaching upward and loaded (trial 8).

Although there is some variability in the contraction level of both muscles according to arm position and wrist load, it is still possible to set a threshold for both muscles above which an intentional control signal would be detected at all of the arm positions and loading conditions tested here. For example, a threshold for the ECR could be set at 0.2. Signals above this level would indicate an intentional control signal and signals below this level would be taken as noise.

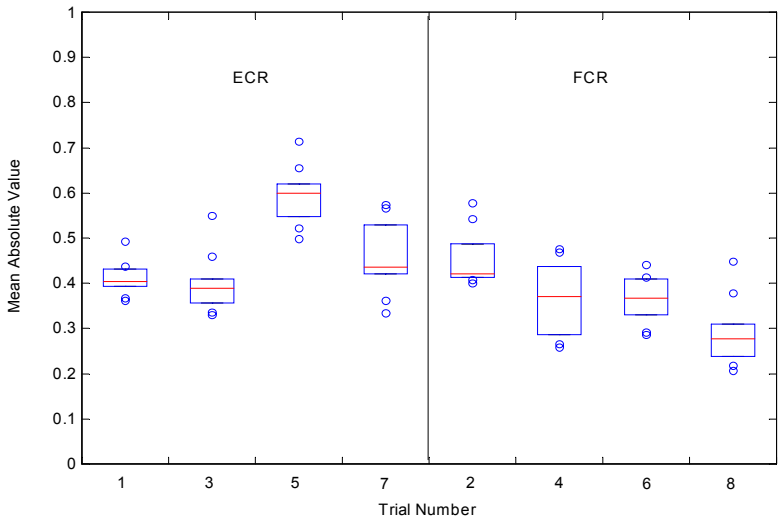

Figure 16. Ranges of contraction levels for each trial from the second protocol. The level of contraction varies with load and position of the arm in space.

Next Quarter

Plans for next quarter include further development of these protocols. Trials for assessing the ability of a subject to produce more distinguishable levels of signal will be added. Data analysis methods for defining thresholds, quantifying repeatability, and quantifying signal distinguishability will be developed. The protocols tested on several able-bodied subjects and the analysis methods will be applied to the resulting data.

Feed-forward ANN-based real-time controller for a hand/wrist neuroprosthesis

Contract section: E.2.a.ii Simultaneous and natural control of multiple arm and hand functions

Abstract

A hand/wrist neuroprosthesis can restore hand grasp and wrist position control in individuals with spinal cord injury at the C5 or C6 level. We created a computer-based real-time controller for such a neuroprosthesis using artificial neural networks (ANNs) to improve muscle-stimulation profiling for SCI individuals.

Both extrinsic finger muscles and wrist muscles produce moments at the wrist, making it difficult to control both hand grasp and wrist position. The ANN-based controller takes into account known interactions to independently control wrist angle, hand opening, and grasp force.

This controller uses high-level programming languages such as LabVIEW and Matlab, which allow the investigator to easily modify the control algorithms while still providing real-time data collection, analysis and muscle stimulation. Several hand/wrist parameters are

measured and analyzed on-line by the controller and used to train the ANN. Stimulation parameters are calculated in real-time, resulting in real-time muscle stimulation.

This controller allows fast changes to control algorithms while fulfilling the performance requirements for neuroprosthesis control. It currently runs on a Pentium III PC under Windows NT, although further development will lead to portable system implementation.

Initial tests were performed in two human subjects. The trained neural networks, with 23-30 hidden neurons, fit the training data well. Pulse width errors were reasonably low. In a few cases, the wrist angle and grasp force errors were also controlled well. However, control was usually poor. Off-line analysis of the data and networks following the experiments showed that the network prediction error was large. When grasp forces and wrist angle that were not part of the training data were applied to the network, the pulse widths were often completely outside of the training data range, and prediction errors were large. Reducing the number of hidden neurons to approximately five, should greatly reduce the prediction error. New tests in human subjects will be carried out to test this hypothesis.

Other potential sources of error during control are voluntary intervention because of residual motor function and muscle fatigue, neither of which are predicted by the network.

Methods

Controller Design

The controller should dynamically adjust muscle stimulation levels to provide independent hand/wrist control. A feed-forward controller was selected because it predicts and compensates for known system interactions. Muscle stimulation is controlled by the neuroprosthesis and therefore, the effect of muscle activation on the wrist position due to electrical stimulation is predictable to some extent. An ANN-based feed-forward controller could model the interaction between hand and wrist muscles, producing muscle stimulation patterns appropriate for a given grasp task.

The controller (Figure 17) consists of two stages: Coordination Templates and Interaction Networks.

1. The coordination templates specify desired grasp and wrist posture based on a grasp command and a grasp mode. The inputs to the coordination network are grasp mode (palmar, lateral, etc.), grasp command level (0-100%), wrist angle, and forearm orientation in the gravitational field. The outputs of the coordination template are the desired grasp opening and force.

2. The interaction networks specify the stimulation pulse-widths that will be delivered to each muscle in order to produce the desired posture specified by the coordination template. Three modules (wrist, thumb, and fingers) compose the interaction networks. Each module controls a different set of muscles. The inputs to the interaction networks are the outputs from the coordination templates, and its outputs are the muscle stimulation levels required to achieve the hand posture specified by the coordination template. The interaction networks were implemented with multiple-layer backpropagation ANNs using MATLAB© Neural Network Toolbox©.

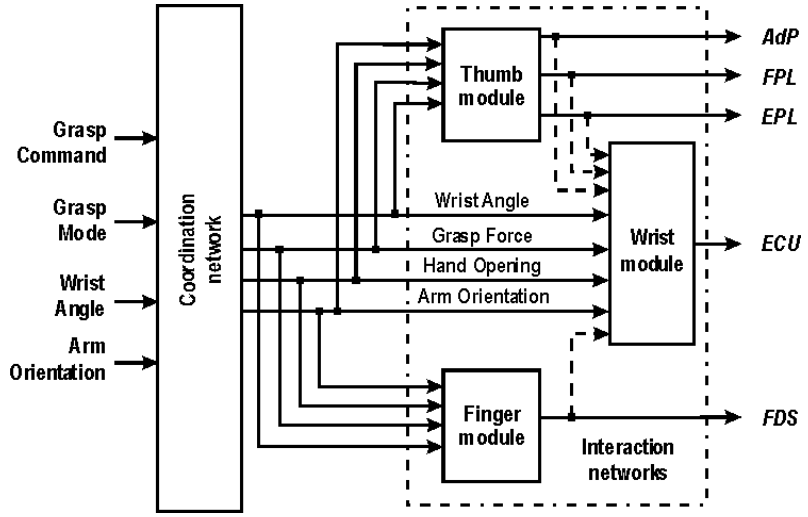

Figure 17. Feed-Forward Controller

Software design and implementation

The system software was divided in two phases: system training and real-time control of the hand/wrist.

1. During the training phase, information about the hand motor system was measured and used to train the neural network (Figure 18). Muscles were stimulated at 16Hz using 50-60 different combinations of pulse-widths for 5-seconds each, followed by a 3-seconds rest interval. Forearm orientation, wrist angle, grasp force, and grasp opening were recorded during muscle stimulation. Custom software written in LabVIEW© analyzed the recorded raw data and calculated the average values of grasp output and wrist angle over the last two seconds of stimulation at each combination of pulse-widths. These steady-state values comprise the data set used to train the neural network. The inputs to the ANN

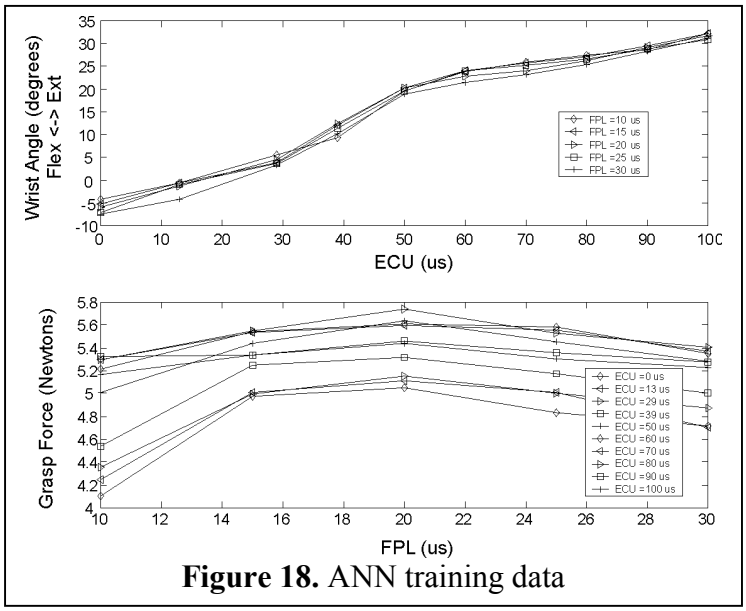

Figure 18. ANN training data

were the muscle responses to stimulation, while the stimulation pulse-widths were used as the ANN targets.

2. The control phase of the system software was designed to run in real-time. An external pulse train generated by a timer in the data acquisition card provided real-time synchronization. Each pulse triggered a new software cycle in the control phase. Each cycle consisted of three main operations: input sampling, calculation of stimulation parameters, and control of the stimulator. The software measured grasp command and forearm orientation, and used the trained ANNs to do a feed-forward estimation of the stimulation pulse-widths required for the appropriate muscle activation. The controller then sent the stimulation commands for all the stimulation channels to the stimulator through the computer serial port.

These two phases were implemented using a combination of the high-level languages Matlab and LabVIEW.

Subjects

Two male subjects in their early thirties with C6 SCI participated in the experiments. Both subjects had received an IRS-8 implant and tendon transfers: stimulated ECU to ECRB and voluntary Br to ECRB.

One of the subjects had weak thumb flexors, whereas the other had weak wrist extensors in the presence of stimulation to thumb flexor muscles.

Experimental Protocol

All protocols were approved by the IRB at MetroHealth Medical Center, where the tests were carried out. All subjects signed consent/release forms.

Lateral grasp was tested, for which FDS, FPL, AdP, and ECU muscles were stimulated at 16Hz with 20mA pulses. Commands were controlled both by the investigator (using computer-generated commands) and by the subject (using commands generated by a shoulder-position transducer).

Forearm supination/pronation was restricted with an open cast, and the subjects were instructed to avoid voluntary movements of their wrists and forearms.

The experiments were divided into training and real-time template testing.

1. Training. Muscle stimulation was stepped through several combinations of pulse-widths (5s each) while the data used to generate the ANN training patterns was measured. This data was then normalized and used to train the neural network.

2. Real-time test. The kinematic and kinetic responses to stimulation were measured, and used to assess the match between desired (determined by test templates) and real (measured) angles and forces, for both computer-generated and subject-generated commands.

Results

System Integration

The combination of hardware and software (Figure 19) that was selected provided an adequate, stable and repeatable response time and stimulation frequency.

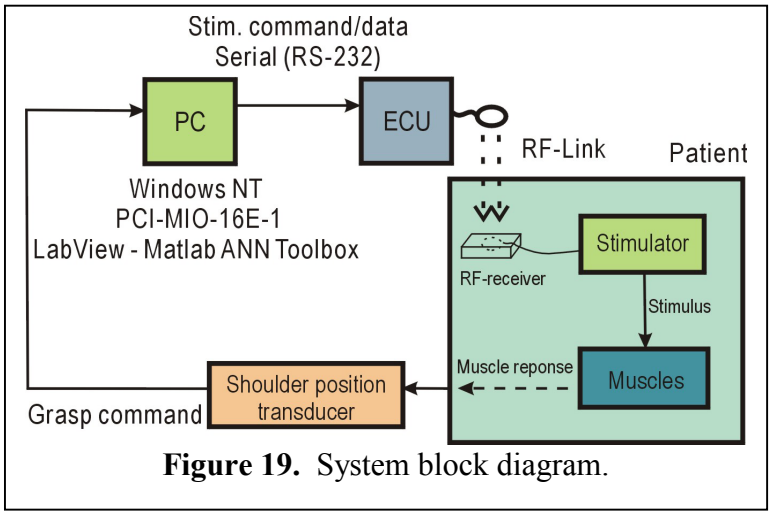

Figure 19. System block diagram.

The high-level development tools provide a flexible environment for faster development. This will also allow future system migration into smaller devices, such as wearable computers.

Real-time system response

The PC-based controller was able to control muscle stimulation in real-time, at up to a

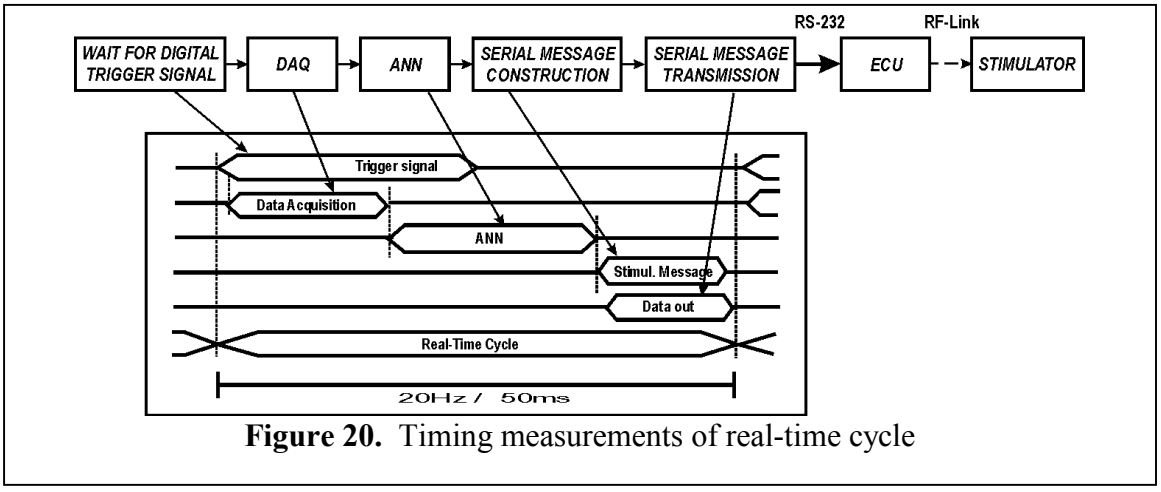

Figure 20. Timing measurements of real-time cycle

24Hz stimulation rate (typical stimulation rate for a neuroprosthesis is 12-16Hz).

Execution time for the real-time control phase was measured using a 20Hz stimulation frequency (T=50ms). The results are shown in Figure 20. Proper stimulus appears at the stimulator output for frequencies up to 24Hz.

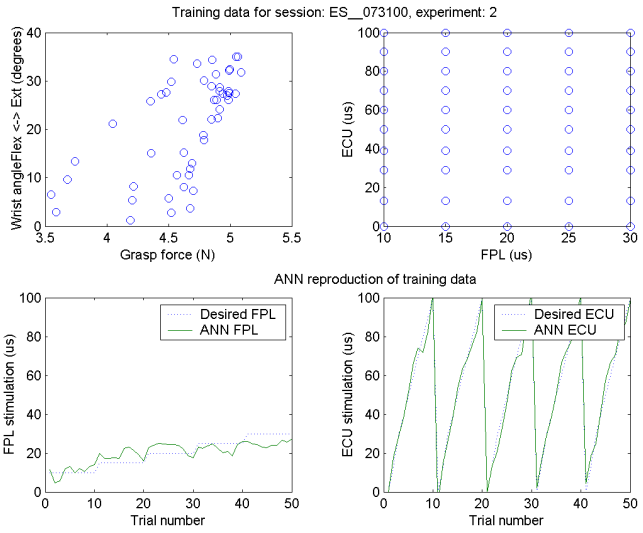

Figure 21. Training data (above) and ANN fitting of the training data (below). In this test, the ECU and FPL were stimulated at several levels, as shown by the dashed lines in the lower plots, and by the circles in the plot at the upper right. The average wrist angle and grasp force at each combination of pulse widths, plotted at the upper left, were used as training data. The trained neural network produced the pulse widths shown by continuous lines in the lower plots. The fitting error was relatively low, i.e., the pulse widths generated by the ANN were close to the actual pulse widths used to obtain the training data.

Training Phase

Neural networks with 23 and 30 hidden neurons were trained using sets of 50 to 60 input-output data patterns. These networks reproduced the training data with small fitting errors (Figure 21).

However, as shown in Figure 22, the neural networks did not always predict pulse-widths within the training data range (poor generalization).

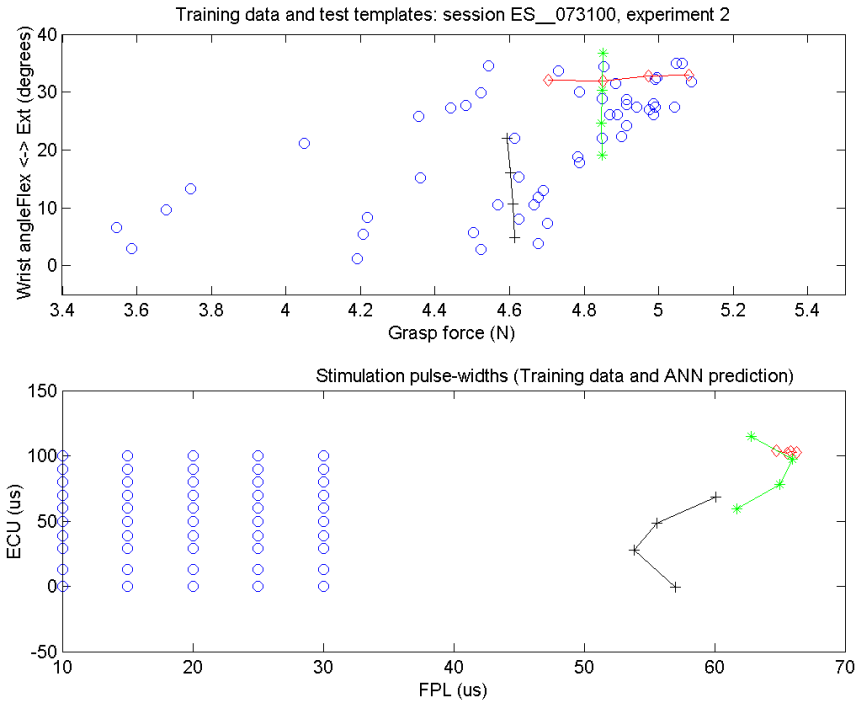

Figure 22. ANN pulse-widths (red, green and black lines in the lower plot) predicted for the selected templates (matching lines in the upper plot) are outside the range of training data, which are shown as circular symbols in both plots. The stimulus pulse widths predicted by the network were entirely outside the range of pulse widths used to obtain the training data, even though the templates were within the range of training data.

Real-Time control phase

The system was effective in controlling muscle stimulation rate in real-time, but in most cases, the hand grasp and wrist motor system did not perform as expected. On some occasions, hand grasp and wrist control were successful (Figures 23 and 24). There were always errors, as would be expected because of the time varying properties of muscles. In some cases the errors would probably be acceptable, but in others, the errors would probably be judged to be excessive.

In these two subjects, the range of hand grasp forces and wrist angles was limited, preventing operation over a wide range of conditions. For some of the grasp templates tested, the ANN was able to predict the stimuli required to achieve the desired wrist angle, but failed to predict those stimuli required to obtain the desired grasp force. In some cases, the desired grasp forces were obtained but only for certain wrist angles. Declining muscle strength (fatigue) also complicated network evaluation.

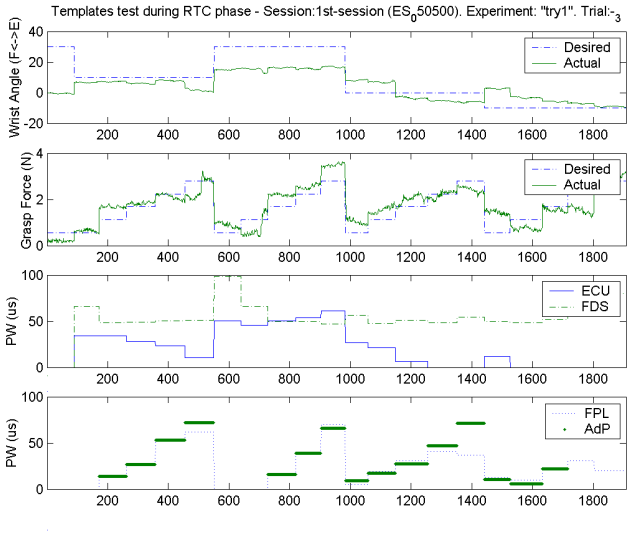

Figure 23. Grasp force resulting from stimulation with predicted pulse-widths matches that of the desired template. Wrist angle shows errors compared to the desired values.

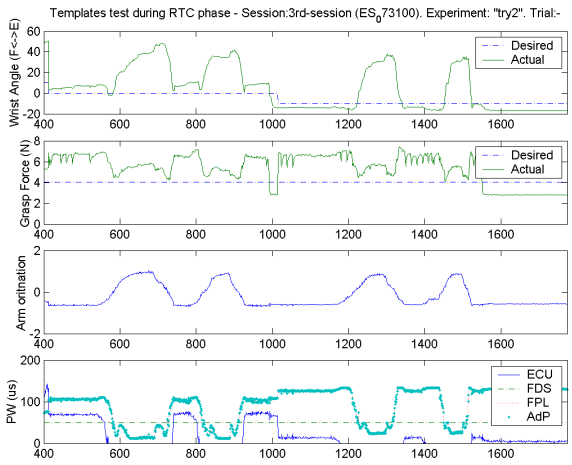

Figure 24. Wrist angle is held constant (at different values) while the subject voluntarily pronates his forearm. No wrist flexors were stimulated. Thus, it is impossible to independently control grasp force (which also produces wrist flexion) And wrist angle when the forearm was supinated.

Off-line ANN simulations

Increasing the range and size of the training data improved generalization. For example, even with a large number of neurons, incorporating additional data resulted in a network that predicted pulse widths within the range of the training data, Figure 25.

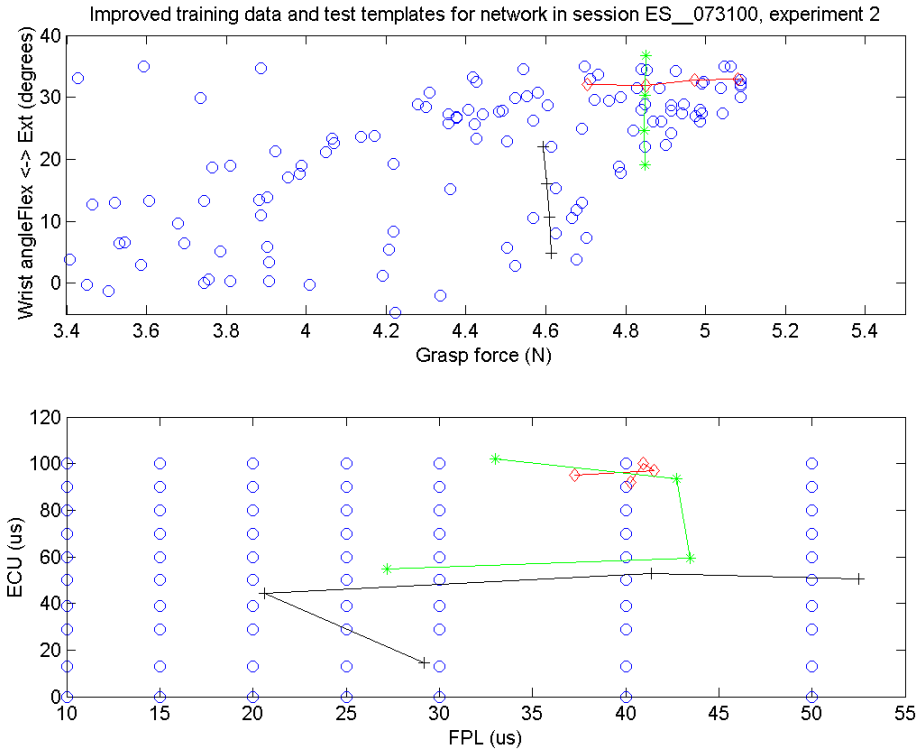

Figure 25. When training data was increased in size and range, the predicted pulse-widths were within the training data range.

Training smaller networks with the same experimental data used on the larger networks resulted on similar fitting errors but significantly lower prediction (generalization) errors (Figure 26). In this case, the larger data sets were studied. Training was done with one subset of the data (80% of the points randomly chosen from the complete set) and fitting error was computed for these inputs. The remainder of the data was used to compute the prediction error, thereby using data that was not part of the training data set.

Figure 27 shows that prediction error decreases with the number of neurons, but it increases again when the number of hidden neurons is too small. The network does not have sufficient learning capacity when it contains less than 4 hidden neurons.

Next quarter

We are intending to perform another human experiment to measure network performance with more training data and the new, smaller network structure and thus verify the findings of the off-line simulations.

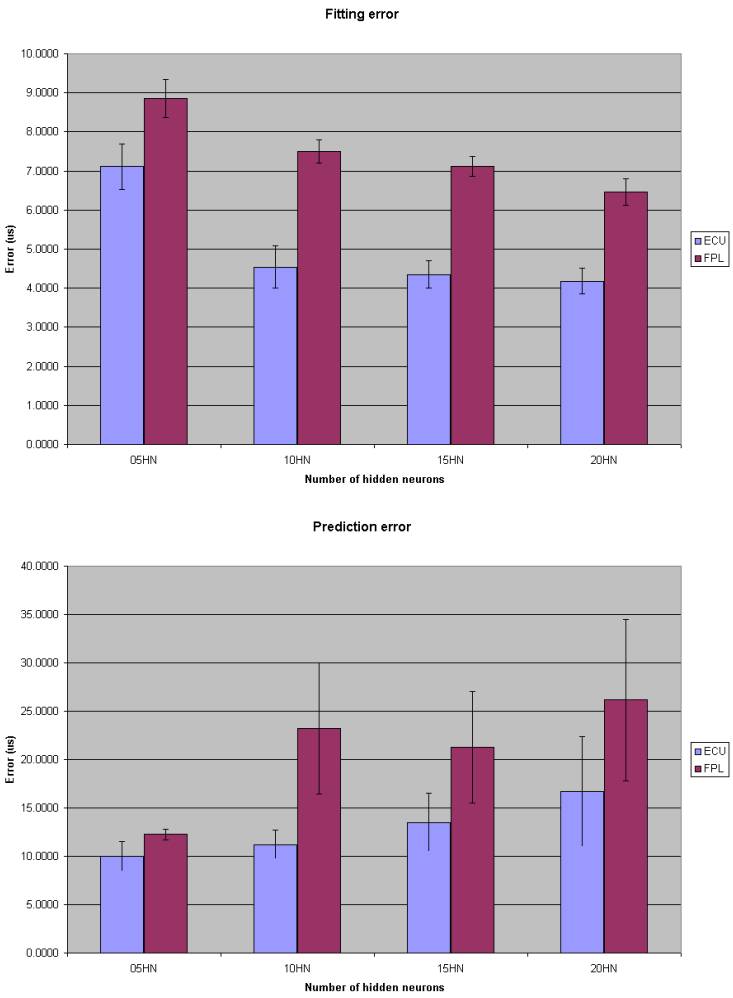

Figure 26. As the number of hidden neurons increases, so does the prediction error, even though the fitting error decreases.

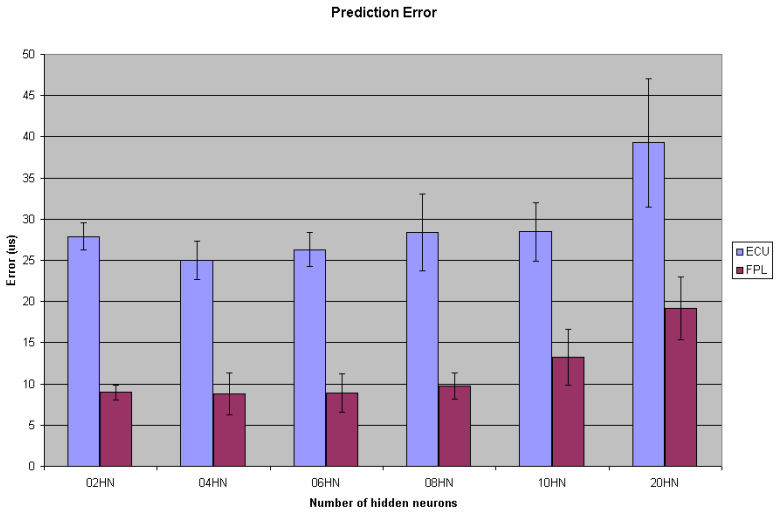

Figure 27. Prediction error as a function of the number of neurons in the hidden layer.

References

Adamczyk, M; Crago, P.E. "Simulated feed-forward neural network coordination of hand grasp and wrist angle in a neuroprosthesis" *IEEE Trans. on Rehab. Eng*, pp. 297 –304, 2000.

Buckett, J.R. "A flexible portable system for Neuromuscular Stimulation in the Paralyzed Upper Extremity". *IEEE Tran. On Biomed. Eng,* vol. 35, No. 11, pp. 897-904. November 1988.

Demuth, H; Beale, M. *Neural Network Toolbox for use with MATLAB*, The Mathworks, Inc. p 138, 1992.

Khanna, T. *Foundations of Neural Networks*. Addison-Wesley, 1990.

Stein, R.B; Peckham, P.H; Popovic, D.P. *Neural Prostheses: Replacing Motor Function After Disease or Disability*. New York: Oxford University Press, 1992.